

Amendments to the Claims

Please amend claims 1, 5 and 9 as shown in the following list of claims.
This listing of claims will replace all prior versions, and listings, of claims in the
5 application.

1 1. (currently amended) Computer graphics processor having a renderer for
2 rendering in parallel N views of 3D images, N, 3D images of a 3D model, said
3 renderer comprising:
4 a rasterizer configured to transverse ~~for transversing~~ a surface grid
5 over a surface of a primitive of a 3D image ~~primitives of said 3D images~~ for all N
6 different views of said 3D image such that transversing is performed once for said
7 3D image, images;
8 a shader unit configured to determine ~~for determining~~ a color of the
9 output of the rasterizer ~~rasteriser~~ and forward ~~forwarding~~ a shaded color sample
10 along with its screen coordinates, and
11 N screen space ~~resamplers~~ resamplers, each of said screen space
12 resamplers being configured to resample ~~each for resampling~~ the shaded color
13 sample determined by said shader unit according to one of the N different views
14 such that resampling is performed N times in parallel for said 3D image.

1 2. (previously presented) Computer graphics processor according to claim 1,
2 further comprising:
3 a texture memory for storing texture maps,
4 wherein said surface grid is derived from a texture map being
5 associated with said primitive and being stored in said texture memory.

1 3. (previously presented) Computer graphics processor according to claim 2,
2 wherein a grid associated to one of the texture maps stored in the
3 texture memory is chosen as said surface grid, if three requirements are fulfilled,
4 said three requirements including:
5 said texture map is addressed independently,
6 said texture map is based on a 2D texture, and

7 the texture coordinates at the vertices do not make up a degenerate
8 primitive.

1 4. (previously presented) Computer graphics processor according to claim 3,
2 wherein

3 the texture map with the largest area in texture space is chosen, if
4 more than one texture maps stored in said texture memory fulfill said three
5 requirements.

1 5. (currently amended) Computer graphics processor according to claim 1 or
2 2, further comprising:

3 a means for addressing a display screen,

4 said renderer having an input for [[the]] a 3D model and an input
5 for at least one viewpoint for rendering image information for supplying to the
6 addressing means,

7 wherein the renderer further comprises an initial part having an
8 input for the 3-D model and for at least one main view point for rendering objects
9 in the form of at least one main view point Z-stack having stack layers with color
10 information and Z-values,

11 the renderer further comprising

12 a Z-stack constructor in which, from the at least one main view
13 point Z-stack generated by the initial stage, Z-stacks for additional viewpoints are
14 constructed, and a further image information occlusion semantics stage for
15 generating image information from the z-stacks.

1 6. (previously presented) Computer graphics processor according to claim 5,
2 wherein said renderer further comprises

3 an object extractor for extraction of objects from a view point z-
4 stack.

1 7. (previously presented) Computer graphics processor according to claim 6,
2 wherein the object extractor is arranged for extracting objects from the at least one
3 main view point z-stack.

1 8. (previously presented) Computer graphics processor according to claim 5,
2 wherein the renderer comprises a DOF rendering stage
3 wherein the DOF rendering stage is arranged for DOF processing
4 of the at least one main view point z-stack into at least one main view point z-
5 stack comprising DOF blurring.

1 9. (currently amended) Method of rendering N different views of 3D images,
2 comprising the steps of:
3 transversing a surface grid over a surface of a primitive of a 3D
4 image primitives of said 3D images for all the different N views of said 3D image
5 such that the transversing is performed once for said 3D image, 3D images,
6 determining a color of the output of the transversing and
7 forwarding a shaded color sample along with its screen coordinates, and
8 resampling the shaded color sample for each of the N different
9 views such that the resampling is performed N times in parallel for said 3D image.

1 10. (previously presented) Method of rendering N views of 3D images
2 according to claim 9, further comprising the steps of:
3 storing texture maps in a texture memory
4 wherein said surface grid is derived from a texture map being
5 associated with said primitive and being stored in said texture memory.

1 11. (previously presented) Method of rendering N views of 3D images
2 according to claim 10,
3 wherein a grid associated to one of the texture maps stored in the
4 texture memory is chosen as surface grid, if three requirements are fulfilled, said
5 three requirements including:
6 said texture map is addressed independently,
7 said texture map is based on a 2D texture, and
8 the texture coordinates at the vertices do not make up a degenerate
9 primitive.

1 12. (previously presented) Method of rendering N views of 3D images
2 according to claim 11, wherein
3 the texture map with the largest area in texture space is chosen, if
4 more than one texture maps stored in said texture memory fulfill said three
5 requirements.

1 13. (previously presented) Method of rendering N views of 3D images
2 according to claim 11, further comprising the steps of:
3 supplying data and addressing means of a 3D display device
4 wherein for a main view point objects in the form of at least one main view point
5 Z-stack comprising stack layers are rendered with RGB and Z-values, and
6 constructing from the at least one main view point Z-stack z-stacks
7 for additional viewpoints, and
8 generating from the Z-stacks for additional viewpoints by means of
9 Z-tracing data to be supplied to the addressing means.

1 14. (previously presented) Computer program product comprising program
2 code means stored on a computer readable medium for performing a method
3 according to claim 9, when said program is run on a computer.